

**In the Specification**

Please amend the paragraph beginning at page 3, line 4, as follows:

Linking between companies is still usually done on a one to one basis between the companies. It is necessary to establish a relationship between two companies, and determine a protocol whereby data may be shared between the companies and messages passed. This is relatively inefficient and redundant, and results in numerous companies creating the same or similar subsystems multiple times. It would be desirable to provide a central clearing house, or exchange, by which companies can communicate to each other. It would be desirable for such exchange to provide additional functionality and intelligence beyond merely passing ~~messages~~ messages between companies using the exchange.

Please amend the paragraph beginning at page 10, line 19, as follows:

~~Conceptually~~ Conceptually, handling of messages within the exchange is broken into three parts. An event container 30 accepts incoming events (messages) 32, and stores date and time information about them. As described below, ~~the~~ event container 30 preferably contains a timer 34, enabling time sensitive events to be handled. A condition container 36 contains instances of conditions 38, which are generally supplied by users of the system. An action container 40 contains instances of actions 42, which are also generally supplied by users.

Please amend the paragraph beginning at page 13, line 22, as follows:

Referring to Figure 4, an event framework ~~se~~ 50 is an operational portion of the event container. It contains a listener, which constantly scans for events. The timer 34 generates timing events, 52, 54 which are recognized by the event framework 50. In other words, the event framework is aware of the current time. In the present case, incoming events are only to be processed between the hours of 8:00 a.m. and 6:00 p.m.

Please amend the paragraph beginning at page 17, line 8, as follows:

Referring to Figure 6, a simple petri-net which corresponds to the conjunction of two events generating an action is shown. The first and second event, represented by ~~circles 70~~ circles 70 and 72, correspond to "places" in petri-net terminology. Action event 74 also corresponds to a place. The condition instance 76 corresponds to a transition. In Figure 6, the transition occurs if both the first and second events have occurred, causing the resulting action 74 to be generated. The first two places correspond to events received by the event container, and the resulting place corresponds to an event generated by an action instance.

Please amend the paragraph beginning at page 18, line 13, as follows:

Figure 9 is a more complex petri-net representing a condition similar to the request for three quotes described above. In this set of conditions, the customer desires to make a selection only when three separate quotes have been submitted in response to a request. When each of the quotes Q1, Q2, and Q3 have been submitted, a transition occurs which generates two ~~outputs 82, 84~~ output actions 84, 86. The first output action ~~82~~ 84 is an acknowledgement to all who have submitted quotes that the quotes have been received, and the second output action ~~84~~ 86 is submission of the quotes to a selection process. This may be automated, or may be reported to a person to make decision as to which quote is to be accepted. If selection is automated, the selection may be as complex as necessary. The selector second output action 84, 86 represents activity which may take place out of the exchange, by sending appropriate messages to the company which will be returned when a decision has been made. Once a decision has been made and returned to the exchange, the selector place 84 will be filled by a token, which will initiate the second transition. Outputs from the second transition are, in this example, to enter an order with the company providing the winning quote ~~86~~ 88, and to send a notice of non-acceptance to the others ~~88~~ 90.

Please amend the paragraph beginning at page 19, line 6, as follows:

It will be appreciated that this petri-net corresponds to the logic of the exchange controller. Quotes Q1, Q2, and Q3 correspond to messages sent to the exchange in response to a bid. The condition defined by the customer requires three quotes to be submitted before a decision is made, so that acknowledgement of the submissions and initiation of the selection process, are made only after three quotes are received. As described above, counting three quotes is ~~preferably~~ preferably done in the event container, but could be implemented in the condition container if desired. The selection process can ~~be a~~ be as simple or as complex as desired by the customer, and can be entirely automated or entirely manual. Once the customer makes a selection, an event is sent to the exchange which corresponds to the selection node. This triggers a second condition instance, which generates the order and sends a notice of non-acceptance to the losing bids.